Combinatorial Probability and the Tightness of Generalization Bounds

K. V. Vorontsov

Dorodnicyn Computing Centre, Russian Academy of Sciences, ul. Vavilova 40, Moscow, 119333 Russia
e-mail: voron@ccas.ru

Abstract—Accurate prediction of the generalization ability of a learning algorithm is an important problem in computational learning theory. The classical Vapnik–Chervonenkis (VC) generalization bounds are too general and therefore overestimate the expected error. Recently obtained data-dependent bounds are still overestimated. To find out why the bounds are loose, we reject the uniform convergence principle and apply a purely combinatorial approach that is free of any probabilistic assumptions, makes no approximations, and provides an empirical control of looseness. We introduce new data-dependent complexity measures: a *local shatter coefficient* and a nonscalar *local shatter profile*, which can give much tighter bounds than the classical *VC shatter coefficient.* An experiment on real datasets shows that the effective local measures may take very small values; thus, the effective local VC dimension takes values in [0, 1] and therefore is not related to the dimension of the space.

DOI: 10.1134/S1054661808020090

The number of observations in data analysis problems is always finite. Nevertheless, the widely used concept of *probability* is defined in terms of either an infinitely large sample or a probability measure, which are both unknown in practical situations. A.N. Kolmogorov pointed out that "getting rid of redundant probabilistic assumptions wherever possible seems to be an important problem. In my lectures, I repeatedly stressed the independent value of a purely combinatorial approach to information theory" [1]. In the preface to the book [2], Yu.K. Belyaev wrote, "there is a strong belief that in the theory of sampling methods one can obtain meaningful analogs of the majority of basic statements of probability theory and mathematical statistics, that by now have been obtained under the assumption that the observations are independent." Thus, an idea has long been growing that one can construct a meaningful theory that would deal only with finite samples and would not be based on a hypothetical set of "all conceivable objects" almost none of which has ever been or will be observed in experiment.

In Section 1, we propose a weak probabilistic axiom that is not based on measure theory and in which all probabilities can be estimated directly in experiment. This axiom agrees with the strong (Kolmogorov's) axioms; however, its application is restricted to the problems of data analysis. In Section 2, we introduce a general statement of empirical prediction problem. In Section 3, we consider a problem of predicting the frequency of an event, which is closely related to the law of large numbers. In Section 4, we consider a problem of learning from examples and the Vapnik–Chervonenkis theory (VCT) under the weak axiom. In Section 5, we analyze the main factors that are responsible for overestimation of bounds in the VCT and propose an empirical technique for measuring the degree of overestimation due to each of these factors. In Section 6, we construct a theory of generalization ability for logical rules and consider an algorithm that searches rules in the form of conjunctions. In Section 7, we present the results of empirical measurement of the VCT bounds overestimation for a set of real classification problems from the UCI repository.

1. WEAK PROBABILISTIC AXIOM

Suppose given a set of objects $\mathbb{X}$. Finite sequences of objects are called *samples from* $\mathbb{X}$. Denote the set of all samples from $\mathbb{X}$ by $\mathbb{X}^*$. In any experiment, irrespective of past or future observations, one can observe only a finite set of objects. Therefore, we will consider a sample $X^L = (x_1, \ldots, x_L)$, called a *general* or *full* sample of length L. Denote by S_L the group of all $L!$ permutations of L elements.

Axiom 1.1 (on the independence of elements of a sample). *All permutations of a general sample τX^L, $\tau \in S_L$, have equal chances to realize.*

Definition 1.1. Suppose given a predicate ψ: $\mathbb{X}^* \longrightarrow \{0, 1\}$ on a set of samples. The part of permutations τX^L for which the predicate is true is called the *probability* of event ψ,

$$\mathrm{P}_\tau \psi(\tau X^L) = \frac{1}{L!} \sum_{\tau \in S_L} \psi(\tau X^L). \qquad (1.1)$$

Received December 3, 2007

This probability depends on the sample X^L. We assume that only the sequence in which objects arise, rather than the objects themselves, are random. The symbol P_τ of probability should be interpreted as a short notation of the average over all permutations τ. In the weak axiom, the term *probability* is meant only in this sense—as a synonym for "part of permutations of a sample."

Definition 1.2. Let ξ: $\mathbb{X}^* \longrightarrow \mathbb{R}$ be an arbitrary real function of a sample. A function F_ξ: $\mathbb{R} \longrightarrow [0, 1]$ of the form

$$F_\xi(z) = \mathrm{P}_\tau[\xi(\tau X^L) < z] \qquad (1.2)$$

is called a *distribution* of ξ on the sample X^L.

Definition 1.3. The *mathematical expectation* of a function ξ: $\mathbb{X}^* \longrightarrow \mathbb{R}$ on the sample X^L is an average over all permutations τ:

$$\mathrm{E}_\tau\xi(\tau X^L) = \frac{1}{L!}\sum_{\tau \in S_L} \xi(\tau X^L). \qquad (1.3)$$

Note that the probability and the mathematical expectation are defined formally identically, as the arithmetic mean: $\mathrm{P}_\tau \equiv \mathrm{E}_\tau \equiv \frac{1}{L!}\sum_{\tau \in S_L}$.

Consider an important particular case when the predicate ψ is a function of two subsamples: $\psi(X^L) = \varphi(X^l, X^k)$, where $X^l \cup X^k = X^L$, $l + k = L$, and the value of the predicate φ does not depend on the order of elements in the subsamples X^l and X^k. Consider the set of all $N = C_L^l$ partitions of the general sample X^L into two subsamples X_n^l and X_n^k, where the subscript $n = 1, \dots, N$ denotes the partition number. Then, Axiom 1.1 implies that all partitions have equal chances of realize, and the probability is defined as a part of partitions of the sample X^L:

$$\mathrm{P}_\tau\psi(\tau X^L) = \mathrm{P}_n\varphi(X_n^l, X_n^k) = \frac{1}{N}\sum_{n=1}^{N} \varphi(X_n^l, X_n^k).$$

Comparison with the strong probabilistic axiom. In the classical (Kolmogorov's) theory of probability, a probability space $\langle \mathbb{X}, \Omega, \mathrm{P}\rangle$ is introduced on the set of objects $\mathbb{X}$; here, Ω is an additive σ-algebra of events on $\mathbb{X}$, and P is the probability measure that is defined on elements of Ω and, as a rule, is unknown. One considers samples of objects drawn independently from P and analyzes some measurable functions of these samples.

Under the weak axiom, the probability measure is introduced on a finite set of partitions, the probability distribution being uniform. Nevertheless, these weak probabilistic assumptions are sufficient for obtaining many fundamental facts of probability theory and mathematical statistics.

If probability (1.1) is calculated under the weak axiom, $\mathrm{P}_\tau\psi(\tau X^L) = p(X^L)$, then the result can easily be translated into the strong axiomatics. Indeed, if we assume that the sample X^L is independent, then $\mathrm{P}_{X^L}\psi(X^L) = \mathrm{P}_{X^L}\psi(\tau X^L)$ for any permutation τ; hence,

$$\mathrm{P}_{X^L}\psi(X^L) = \mathrm{E}_{X^L}\mathrm{P}_\tau\psi(\tau X^L) = \mathrm{E}_{X^L}p(X^L).$$

The translation is performed by taking the mathematical expectation E with respect to the sample X^L of the probability $p(X^L)$ obtained. When this probability does not depend on the sample X^L, the result is translated directly. Thus, the correspondence principle holds: two theories lead to the same results whenever both are applicable.

In the strong axiomatics, the distribution functions and mathematical expectations are unobservable: they are expressed either through the passage to the infinite sample or in terms of the probability measure P, which both are unknown in practical situations. Under the weak axiom, one exclusively considers statistics—functions of finite samples z: $\mathbb{X}^* \longrightarrow Z$. In data analysis, the estimation of unobservable quantities seems to be an artificial problem, detached from practice.

2. PROBLEMS OF EMPIRICAL PREDICTION

A problem of empirical prediction consists in the following: having obtained a data sample, predict certain properties of similar data that will be known later and estimate the accuracy of prediction.

Suppose given a set R and a function T: $\mathbb{X}^* \times \mathbb{X}^* \longrightarrow R$. Consider an experiment in which one of equiprobable partitions of the sample X^L into two subsamples X_n^l and X_n^k, $n = 1, \dots, N$, is realized. After the realization of a partition n, an observer is communicated to a subsample X_n^l. Without knowing the hidden subsample X_n^k, the observer must predict the value of $T_n = T(X_n^k, X_n^l)$ that essentially depends on X_n^k. One should also estimate the confidence of the prediction, i.e., the probability that the unknown true value of T_n does not strongly differ from the prediction made.

Problem 2.1. Construct a *predicting function* $\hat{T}$: $\mathbb{X}^* \longrightarrow R$ whose value $\hat{T}_n = \hat{T}(X_n^l)$ on the observed subsample X_n^l approximates the unknown true value $T_n = T(X_n^k, X_n^l)$ and estimate the confidence of the prediction by providing a nonincreasing *bound function* $\eta(\varepsilon)$ such that

$$\mathrm{P}_n[d(\hat{T}_n, T_n) \geq \varepsilon] \leq \eta(\varepsilon), \qquad (2.1)$$

where $d: R \times R \longrightarrow \mathbb{R}$ is a given function characterizing the deviation $d(\hat{T}_n, T_n)$ of the predicted value $\hat{T}_n$ from the unknown true value T_n.

The parameter ε is called the *accuracy*, and the quantity $(1 - \eta(\varepsilon))$ is the *confidence* of the prediction. If equality holds in (2.1), then $\eta(\varepsilon)$ is called an *exact bound.* The bound $\eta(\varepsilon)$ may depend on l and k, as well as on the functions T and $\hat{T}$. Usually, it is assumed that $\varepsilon > 0$ and $0 < \eta < 1$. An empirical prediction is consistent if (2.1) holds for sufficiently small ε and η.

Remark 2.1. If the function $T(U, V)$ depends only on U, then we will omit the second argument V. In some problems, one sets $T(U) = \hat{T}(U)$. Nevertheless, the roles of the functions T and $\hat{T}$ are essentially different. The function T is assumed to be defined a priori and enters the statement of the problem, whereas the prediction function $\hat{T}$ can be chosen by an observer on his own will.

Remark 2.2. The prediction of a certain property of a sample on the basis of the properties of another sample is called a *transduction.* It is believed that transduction is a more primitive and restricted form of reasoning than induction. In our case this is not quite so. If one succeeds in obtaining a bound $\eta(\varepsilon)$ that is valid *for any* sample X^L or at least for a wide class of samples, then transduction becomes as general as induction.

Examples of problems of empirical prediction. Choosing a set R, functions T, $\hat{T}$, and a d, one can obtain the statements of various problems of probability theory, mathematical statistics, and machine learning.

Problem 2.2 (estimation of the frequency of an event). Let $S \subseteq \mathbb{X}$ be a set of objects; we call it an event. Introduce a function of the *frequency of event S* on a finite sample U:

$$\nu_S(U) = \frac{1}{|U|}\sum_{x \in U}[x \in S], \quad U \in \mathbb{X}^*.$$

Set $R = \mathbb{R}$, $T(U) = \hat{T}(U) = \nu_S(U)$, and $d(\hat{r}, r) = |r - \hat{r}|$.

The problem is to predict the frequency of event S on the hidden sample X_n^k by its frequency on the observed sample X_n^l and estimate the confidence of the prediction:

$$\mathrm{P}_n\Big[\big|\nu_S(X_n^k) - \nu_S(X_n^l)\big| \geq \varepsilon\Big] \leq \eta(\varepsilon). \qquad (2.2)$$

Sometimes, it is required to obtain a one-sided, say, an upper, bound. Then one should set $d(\hat{r}, r) = r - \hat{r}$:

$$\mathrm{P}_n[\nu_S(X_n^k) - \nu_S(X_n^l) \geq \varepsilon] \leq \eta(\varepsilon). \qquad (2.3)$$

This problem is of fundamental importance for probability theory and is closely related to the law of large numbers and convergence theorems. Below, we will obtain exact bounds for (2.2) and (2.3). These bounds also arise in practical applications, for example, in sampling quality control [2].

Problem 2.3 (estimation of the distribution function). For an arbitrary function $\xi: \mathbb{X} \longrightarrow \mathbb{R}$ and an arbitrary finite sample $U \in \mathbb{X}^*$, define an *empirical distribution function* $F_\xi: \mathbb{R} \longrightarrow [0, 1]$. This function shows on which part of objects of the sample the value of $\xi(x)$ does not exceed z:

$$F_\xi(z, U) = \frac{1}{|U|}\sum_{x \in U}[\xi(x) \leq z].$$

Take, as R, the set of all nondecreasing piecewise constant functions $F: \mathbb{R} \longrightarrow [0, 1]$. Introduce a uniform distance $d(\hat{r}, r) = \max_{z \in \mathbb{R}}|r(z) - \hat{r}(z)|$ on R. Set $T(U) = \hat{T}(U) = F_\xi(z, U)$.

The problem is to predict the maximal deviation of the distribution $F_\xi(z, X_n^k)$ on a hidden sample from the known distribution $F_\xi(z, X_n^l)$ on an observed sample and estimate the confidence of the prediction:

$$\mathrm{P}_n\Big[\max_{z \in \mathbb{R}}\big|F_\xi(z, X_n^k) - F_\xi(z, X_n^l)\big| \geq \varepsilon\Big] \leq \eta(\varepsilon).$$

This problem is closely related to the convergence of empirical distributions and is of fundamental importance for mathematical statistics. This bound underlies the Smirnov criterion, which is used to testing whether two distributions differ [3, 4]. There is an exact bound for this problem as well; however, its analysis is beyond the scope of the present paper.

Problem 2.4 (learning from examples). Suppose given a set of admissible answers $\mathbb{Y}$. There exists an unknown *target function* $y^*: \mathbb{X} \longrightarrow \mathbb{Y}$ that assigns to each object $x \in \mathbb{X}$ a correct answer $y^*(x)$. A *loss function* $\mathcal{L}: \mathbb{Y} \times \mathbb{Y} \longrightarrow \mathbb{R}$ is defined whose value $\mathcal{L}(y, y')$ characterizes the error of answer y if compared with the correct answer y'. Functions a: $\mathbb{X} \longrightarrow \mathbb{Y}$ admitting an efficient computer implementation are called *algorithms*. The *average error* of a function $a: \mathbb{X} \longrightarrow \mathbb{Y}$ on a finite sample U is given by

$$\nu(a, U) = \frac{1}{|U|}\sum_{x \in U}\mathcal{L}(a(x), y^*(x)), \quad U \in \mathbb{X}^*.$$

Given an observed *training* sample X_n^l with known answers $y_i = y^*(x_i)$, $x_i \in X_n^l$, a *learning algorithm* μ: $\mathbb{X}^* \longrightarrow \mathbb{Y}^{\mathbb{X}}$ constructs a function $a_n = \mu X_n^l$. When the average error on the hidden *testing* sample $\nu(a_n, X_n^k)$ is much greater than the average training error $\nu(a_n, X_n^l)$, it is said that the function a_n is *overfitted* [5, 6].

Introduce the difference between the average errors of the function a_n on two samples:

$$\delta(a_n, X_n^l, X_n^k) = \nu(a_n, X_n^k) - \nu(a_n, X_n^l).$$

Definition 2.1. The difference $\delta(\mu X_n^l, X_n^l, X_n^k)$ between the average errors of a function $a_n = \mu X_n^l$ on the testing and training samples is called the *overfitting* of this function.

Set $R = \mathbb{R}$, $T_n = \nu(a_n, X_n^k)$, $\hat{T}_n = \nu(a_n, X_n^l)$, and $d(\hat{r}, r) = r - \hat{r}$. The problem is to predict the upper bound for the overfitting and estimate the confidence of the prediction:

$$\mathrm{P}_n[\nu(a_n, X_n^k) - \nu(a_n, X_n^l) \geq \varepsilon] \leq \eta(\varepsilon). \tag{2.4}$$

Prevention of overfitting is a central problem in statistical learning theory [7].

Empirical estimation of probability. The results obtained under the weak axiom can always be verified experimentally. Suppose given a set of values $\varphi_n = \varphi(X_n^l, X_n^k)$, $n = 1, \ldots, N$. To estimate the value of the sum

$$Q_N = \mathrm{P}_n\varphi_n \equiv \frac{1}{N}\sum_{n=1}^{N}\varphi_n,$$

we replace the summation over all N partitions by the summation over a certain subset of partitions $N' \subset \{1, \ldots, N\}$, which is large enough to give an accurate estimate but not too large to be computable in a reasonable time:

$$Q_N \approx Q(N') = \hat{\mathrm{P}}_n\varphi_n \equiv \frac{1}{|N'|}\sum_{n \in N'}\varphi_n.$$

For example, in the Monte Carlo method, the subset N' of partitions is chosen randomly and independently of a uniform distribution on $\{1, \ldots, N\}$. In this case, the estimation of the accuracy of approximation $|Q(N') - Q_N|$ reduces to Problem 2.2, except that now one considers partitions as objects.

We will call $\hat{\mathrm{P}}_n \equiv \frac{1}{N'}\sum_{n \in N'}$ an *empirical estimate of probability.*

Empirical estimation has a few significant drawbacks. It requires the knowledge of the full sample X^L and therefore cannot be directly used for empirical prediction. It does not allow one to obtain bounds in the analytic form. Finally, it may require large computational expenditure.

Thus, the applicability domain of empirical estimation is rather limited. In practice, this kind of estimation is used for the experimental investigation of the dependence of Q_N on some parameters of a problem (for example, on the sample length). In problems of learning from examples, empirical estimation is called *cross-validation* and is used for estimating the quality of a learning algorithm μ rather than the quality of an individual function. It is indispensable when theoretical bounds Q_N are either unknown or overestimated. In this paper, empirical estimation is applied to the analysis of the tightness of theoretical bounds.

3. PROBLEM OF ESTIMATING THE FREQUENCY OF AN EVENT

Consider Problem 2.2 on predicting the frequency of event $S \subseteq \mathbb{X}$. Suppose that the number of elements of the event S in the entire sample X^L is fixed and equals $m = L\nu_S(X^L)$. Then the number of elements of S in the observed subsample $l\nu_S(X_n^l)$ and the number of elements of S in the hidden subsample $k\nu_S(X_n^k)$ obey a hypergeometric distribution:

$$\begin{aligned} \mathrm{P}_n[l\nu_S(X_n^l) = s] &= \mathrm{P}_n[k\nu_S(X_n^k) = m - s] \\ &= h\begin{pmatrix} l & s \\ L & m \end{pmatrix} = \frac{C_m^s C_{L-m}^{l-s}}{C_L^l}, \end{aligned} \tag{3.1}$$

where s takes values ranging from $s_0(m) = \max\{0, m - k\}$ to $s_1(m) = \min\{l, m\}$.

Let us introduce contracted notations $\nu_n^l = \nu_S(X_n^l)$ and $\nu_n^k = \nu_S(X_n^k)$.

Theorem 3.1. *The following exact bounds hold for any* $\varepsilon \in [0, 1)$:

$$\mathrm{P}_n[\nu_n^l \leq \varepsilon] = H_L^{l,m}(\lfloor \varepsilon l \rfloor);$$

$$\mathrm{P}_n[\nu_n^k \geq \varepsilon] = H_L^{l,m}(\lfloor m - \varepsilon k \rfloor);$$

$$\begin{aligned} \mathrm{P}_n[\nu_n^k - \nu_n^l \geq \varepsilon] &= H_L^{l,m}(s_m^-(\varepsilon)), \\ s_m^-(\varepsilon) &= \left\lfloor \frac{l}{L}(m - \varepsilon k) \right\rfloor; \end{aligned} \tag{3.2}$$

$$\begin{aligned} \mathrm{P}_n[|\nu_n^k - \nu_n^l| \geq \varepsilon] &= H_L^{l,m}(s_m^-(\varepsilon)) + \bar{H}_L^{l,m}(s_m^+(\varepsilon)), \\ s_m^+(\varepsilon) &= \left\lceil \frac{l}{L}(m + \varepsilon k) \right\rceil. \end{aligned} \tag{3.3}$$

Remark 3.1. In the statement of the theorem, $\lfloor z \rfloor$ is the integer part (floor) of a real number z, i.e., the greatest integer *less than or equal to* z. Similarly, $\lceil z \rceil$ is the least integer *greater than or equal to* z. If we change nonstrict inequalities to the strict ones on the left-hand sides, all the bounds remain valid with one reservation: $\lfloor z \rfloor$ should be understood as the greatest integer *less than* z; it differs from the floor only in that $\lfloor z \rfloor = z - 1$ for integer z. Correspondingly, $\lceil z \rceil$ is the least integer *greater than* z, so that $\lceil z \rceil = z + 1$ for integer z.

Proof. The first two inequalities are immediate corollaries to (3.1); therefore, we begin with the proof of (3.2). Let us group all the terms with equal values of $s = l\nu_n^l$ and sum up them separately:

$$\mathrm{P}_n[\nu_n^k - \nu_n^l \geq \varepsilon]$$

$$= \sum_{s = s_0}^{s_1} \mathrm{P}_n\left[\nu_n^l = \frac{s}{l}\right]\left[\frac{m-s}{k} - \frac{s}{l} \geq \varepsilon\right].$$

The greatest integer that satisfies the inequality $\frac{m-s}{k} - \frac{s}{l} \geq \varepsilon$ is given precisely by $s_m^-(\varepsilon)$; therefore, the expression obtained can be rewritten in a shorter form:

$$\mathrm{P}_n[\nu_n^k - \nu_n^l \geq \varepsilon] = \sum_{s = s_0}^{s_m^-(\varepsilon)} \mathrm{P}_n\left[\nu_n^l = \frac{s}{l}\right]$$

$$= \sum_{s = s_0}^{s_m^-(\varepsilon)} h\begin{pmatrix} l & s \\ L & m \end{pmatrix} = H_L^{l,\,m}(s_m^-(\varepsilon)).$$

The two-sided bound (3.3) is proved analogously if we divide the partition set into two disjoint subsets:

$$\mathrm{P}_n[|\nu_n^k - \nu_n^l| \geq \varepsilon] = \mathrm{P}_n[\nu_n^k - \nu_n^l \geq \varepsilon]$$

$$+ \mathrm{P}_n[\nu_n^l - \nu_n^k \geq \varepsilon] = H_L^{l,\,m}(s_m^-(\varepsilon)) + \bar{H}_L^{l,\,m}(s_m^+(\varepsilon)).$$

The theorem is proved.

Upper bound. The number of elements m of event S in the full sample X^L cannot be determined while the hidden part of the data is unknown. At the same time, the bound functions (3.2) and (3.3) depend on this number. The simplest solution of this problem is to take the maximum over m and obtain an overestimated upper bound instead of the exact bound:

$$\mathrm{P}_n[\nu_n^k - \nu_n^l \geq \varepsilon] \leq \max_{m = 0, \dots, L} H_L^{l,\,m}(s_m^-(\varepsilon)) \equiv \Gamma_L^l(\varepsilon). \quad (3.4)$$

Here, it is sufficient to take the maximum over all m starting from $\lceil \varepsilon k \rceil$ to $\lfloor L - \varepsilon l \rfloor$, because the left-hand side of the inequality vanishes for other values of m.

There is a known asymptotic behavior of $\Gamma_L^l(\varepsilon)$ [5]: for any $\varepsilon > 0$,

$$\Gamma_L^l(\varepsilon) \sim \exp\left(-2\varepsilon^2 \frac{lk}{l+k}\right), \quad l, k \longrightarrow \infty,$$

whence it follows that the probabilities $\mathrm{P}_n[\nu_n^k - \nu_n^l \geq \varepsilon]$ and $\mathrm{P}_n[|\nu_n^k - \nu_n^l| \geq \varepsilon]$ tend to zero as l and k tend simultaneously to infinity. This means that equalities (3.2) and (3.3) represent an analog of the law of large numbers under the weak axiom.

4. PROBLEM OF LEARNING FROM EXAMPLES

Let us refine the statement of Problem 2.4 on predicting the quality of learning from examples. We will consider only binary loss functions, assuming

$\mathscr{L}(y, y')$ = [answer y is erroneous for a correct answer y'].

The choice of a loss function depends on a specific problem, first of all, on the set of admissible answers $\mathbb{Y}$. In classification, $\mathbb{Y}$ is a finite set of classes; then $\mathscr{L}(y, y') = [y \neq y']$. In regression, where $\mathbb{Y} = \mathbb{R}$, it is conventional to use smooth loss functions, like $\mathscr{L}(y, y') = (y - y')^2$. However, one can also introduce a binary loss function: $\mathscr{L}(y, y') = [|y - y'| \geq d]$, where d is a fixed threshold value.

The form of a binary function is unimportant for further analysis. The main results are valid for a wide class of problems, including both classification and regression.

The Classical Vapnik–Chervonenkis theory [8, 9, 5] (VCT) is based on the Kolmogorov's probabilistic axiomatics. It is assumed that the set of objects $\mathbb{X}$ is a probability space with some unknown probability measure, and that all the samples considered are i.i.d. (independent identically distributed).

Suppose given a set of functions $A = \{a\colon \mathbb{X} \longrightarrow \mathbb{Y}\}$. Among these functions, we choose one a^* that makes the minimum number of errors on a given training sample X^l:

$$a^* = \underset{a \in A}{\operatorname{argmin}}\, \nu(a, X^l).$$

This method is called *empirical risk minimization (ERM)*. There may exist several functions in the set that minimize the empirical risk. It is assumed that any of these functions can be chosen as a solution. Other learning algorithms are not considered in the classical variant of the VCT.

The quality of a function a^* is characterized by the probability of error $P(a^*)$. A sufficient condition of learnability is that the deviation of the empirical error $\nu(a, X^l)$ from its probability $P(a)$ should be small for any $a \in A$. More precisely, the following bound must hold for sufficiently small values of accuracy ε and confidence η:

$$P_\varepsilon(A) = \mathrm{P}\left\{\sup_{a \in A}\left|P(a) - \nu(a, X^l)\right| > \varepsilon\right\} \le \eta(\varepsilon). \quad (4.1)$$

The introduction of the supremum yields a guaranteed bound, which is valid irrespective of what function a^* will be obtained as a result of learning. If the right-hand side of (4.1) tends to zero as l, $k \longrightarrow \infty$, then it is said that the error rate *converges uniformly* to the error probability.

One can also characterize the quality of function a^* by the error rate $\nu(a^*, X^k)$ on an i.d.d. test sample X^k. Then, one gets more accurate bounds in view of the main lemma proved in [9, p. 219] for $l = k$:

$$\begin{aligned}&\mathrm{P}\left\{\sup_{a \in A}\left|\mathrm{P}(a) - \nu(a, X^l)\right| > \varepsilon\right\}\\ &\le 2\mathrm{P}\left\{\max_{a \in A}\left|(a, X^k) - \nu(a, X^l)\right| > \frac{1}{2}\varepsilon\right\}.\end{aligned} \quad (4.2)$$

Later, this bound was refined [5]: $\frac{1}{2}\varepsilon$ on the right-hand side was replaced by $\varepsilon - \frac{1}{l}$.

If the right-hand side tends to zero as l, $k \longrightarrow \infty$, then it is said that the error rates of two samples *converge uniformly.*

It is quite sufficient to take into account only positive deviations of frequencies, because negative deviations $\nu(a, X^k) - \nu(a, X^l) < 0$ testify to good learnability. In this case, the accuracy increases again by a factor of two, and we arrive at a functional of uniform one-sided deviation of frequencies in two samples:

$$P_\varepsilon(A) = \mathrm{P}\{\max_{a \in A}(\nu(a, X^k) - \nu(a, X^l)) > \varepsilon\}. \quad (4.3)$$

When $l = k$, the following bound for the uniform convergence rate is valid for any probability distribution on $\mathbb{X}$ and any target function y^* [9]:

$$P_\varepsilon(A) \le \Delta^A(2l) \cdot \frac{3}{2}e^{-\varepsilon^2 l}, \quad (4.4)$$

where $\Delta^A(L)$ is the growth function of the set of functions A. The growth function is introduced as follows.

Definition 4.1. Functions a and a' are *indistinguishable* on the sample X^L if they make errors on the same objects: $\mathscr{L}(a(x_i), y_i) = \mathscr{L}(a'(x_i), y_i)$ for any $x_i \in X^L$.

Indistinguishability is an equivalence relation on the set A.

Definition 4.2. A *shatter coefficient* $\Delta(A, X^L)$ of the set of functions A on the sample X^L is the number of equivalence classes induced on the set A by the relation of indistinguishability of functions on the sample X^L.

We can reformulate this definition as follows: a shatter coefficient is the number of different binary vectors $[\mathscr{L}(a(x_i), y^*(x_i))]_{i=1}^L$ generated by all possible functions $a \in A$ on a given sample X^L. In problems of classification into two classes, the shatter coefficient is equal to the number of different *dichotomies* (partitionings into two classes) realized by all possible functions from the set A.

Definition 4.3. A *growth function* of the set of functions A is the maximal value of the shatter coefficient $\Delta^A(X^L)$ over all possible samples of length L:

$$\Delta^A(L) = \max_{X^L}\Delta(A, X^L), \quad L = 1, 2, 3, \ldots.$$

The growth function depends neither on the sample nor on the learning algorithm and characterises a complexity of the set of functions A. The upper bound $\Delta^A(L) \le 2^L$ is obvious.

The minimal number h for which $\Delta^A(h) < 2h$ is called the *Vapnik–Chervonenkis dimension* (VC dimension) of the set of functions A. If such an h does not exist, then it is said that the dimension of A is infinite. It was proved that if A has a finite dimension h, then its growth function depends polynomially on L:

$$\Delta^A(L) \le C_L^0 + C_L^1 + \ldots + C_L^h \le \frac{3}{2}\frac{L^h}{h!}. \quad (4.5)$$

In this case, the uniform convergence takes place, and the set A is learnable. Thus, in VCT, to obtain an upper bound of overfitting, it is sufficient to know the sample length and the VC dimension of the set of functions.

The practical application of the approach described is hampered by the fact that bound (4.4) is highly overestimated. To verify this, it suffices to calculate numerically the required length of the training sample l as a function of (h, η, ε). This length is on the order of 10^5–10^9, which is much greater than the number of objects which one usually deals with in practice [7].

The reason why the values of bounds in the VCT are overestimated lies in their extreme generality. They are valid for any probability distribution on $\mathbb{X}$, any target function $y^*(x)$, and any learning algorithm μ. Therefore, the bounds are pessimistically related to the worst case, which is hardly ever encountered in practice.

The introduction of the concept of *learning algorithm* μ makes obvious that even the functional of uniform convergence itself represents an overestimated upper bound:

$$\mathrm{P}\{\delta(\mu X^l, X^l, X^k) > \varepsilon\} \le \mathrm{P}\{\max_{a \in A}\delta(a, X^l, X^k) > \varepsilon\}. \, (4.6)$$

It is the small value of the left-hand rather than the right-hand side of this inequality that one should have taken as the definition of *learnability* from examples. In the VCT, a lot of attention is paid to the necessary and sufficient conditions for uniform convergence. However, according to (4.6), the uniform convergence is only a sufficient condition for learnability. If the VC dimension is infinite and there is no uniform convergence, it is too early to make a conclusion that there is no learnability. A common error in interpreting the VCT is the conclusion that one should restrict the complexity of the set of functions. This conclusion would be correct if the bounds of the VCT were sufficiently exact.

The Vapnik–Chervonenkis theory under the weak axiom. Suppose that, in Problem (2.4) (learning from examples), an algorithm μ yields the same fixed function a for any sample. In this simplified statement, the problem reduces to estimating the frequency of a fixed event $S = \{x \in \mathbb{X} | \mathcal{L}(a(x), y^*(x)) = 1\}$, and (3.4) implies the following proposition.

Proposition 4.1. The following bound holds for any $a: \mathbb{X} \longrightarrow \mathbb{Y}$ and any $\varepsilon \in [0, 1)$:

$$\mathrm{P}_n[\nu(a, X_n^k) - \nu(a, X_n^l) \geq \varepsilon] < \Gamma_L^l(\varepsilon). \qquad (4.7)$$

In this case the law of large numbers applies: the frequency of event S on the test sample can be predicted by its frequency on the training sample, and the accuracy of prediction increases with the sample length.

Now, consider the general case when algorithm μ yields different functions on different training samples. Denote by A_L^l the set of functions generated by the algorithm μ on all possible subsamples $X_n^l \subset X^L$:

$$A_L^l \equiv A_L^l(\mu, X^L) = \{a_n = \mu X_n^l | n = 1, \ldots, N\}.$$

The cardinality of the set A_L^l is no greater than $N = C_L^l$. It may even be less than N if the learning algorithm μ generates identical functions on different subsamples. The shatter coefficient of the set A_L^l may be still less if some functions, which do not coincide as maps $\mathbb{X} \longrightarrow \mathbb{Y}$, are indistinguishable on the sample X^L.

Definition 4.4. The shatter coefficient of the set of functions $A_L^l(\mu, X^L)$ is called a *local shatter coefficient* of the learning algorithm μ on the sample X^L and is denoted by $\Delta_L^l \equiv \Delta_L^l(\mu, X^L) = \Delta(A_L^l(\mu, X^L), X^L)$.

The set of functions A_L^l is divided into $L + 1$ subsets of functions A_m that make a fixed number of errors $m = 0, 1, \ldots, L$ on the general sample X^L:

$$A_m \equiv A_{L,m}^l(\mu, X^L)$$
$$= \left\{a_n = \mu X_n^l \middle| \nu(a_n, X^L) = \frac{m}{L}, n = 1, \ldots, N\right\},$$
$$A_L^l = A_0 \cup A_1 \cup \ldots \cup A_L.$$

Definition 4.5. A sequence of shatter coefficients $D_m \equiv \Delta_{L,m}^l(\mu, X^L) = \Delta(A_{L,m}^l(\mu, X^L), X^L)$, $m = 0, 1, \ldots, L$, is called a *local shatter profile* of the learning algorithm μ on the sample X^L.

The sets A_m are disjoint, and their union yields A_L^l. Therefore,

$$\Delta_L^l = D_0 + D_1 + \ldots + D_L. \qquad (4.8)$$

Recall that, in Problem 2.4 (learning from examples), we are interested in the upper bounds of functional (2.4):

$$Q_\varepsilon \equiv Q_\varepsilon(\mu, X^L) = \mathrm{P}_n[\delta_n \geq \varepsilon],$$

where $\delta_n = \delta(a_n, X_n^l, X_n^k)$ is the overfitting of the function $a_n = \mu X_n^l$.

Let us divide the functional Q_ε into $L + 1$ terms $Q_{\varepsilon,m} \equiv Q_{\varepsilon,m}(\mu, X^L)$ with respect to the parameter m:

$$Q_\varepsilon = \sum_{m=0}^{L} \mathrm{P}_n[\delta_n \geq \varepsilon]\left[\nu(a_n, X^L) = \frac{m}{L}\right] = \sum_{m=0}^{L} Q_{\varepsilon,m}.$$

Theorem 4.2. The following bound holds for any $\varepsilon \in [0, 1)$ and $m = 0, 1, \ldots, L$:

$$Q_{\varepsilon,m} \leq D_m H_L^{l,m}(s_m^-(\varepsilon)). \qquad (4.9)$$

Proof. The indistinguishability relation of functions on the sample X^L divides the set of functions A_m into equivalence classes A_{md}, where $d = 1, \ldots, D_m$ is the index of a class among all D_m classes whose functions make m errors. Let us express P_n in terms of a sum of partitions taken separately for each equivalence class:

$$Q_{\varepsilon,m} = \sum_{d=1}^{D_m} \frac{1}{N} \sum_{n=1}^{N} [a_n \in A_{md}]$$
$$\times [\nu(a_m, X_n^k) - \nu(a_n, X_n^l) \geq \varepsilon].$$

The value of the functional is not changed if we replace the function $a_n = \mu X_n^l$ by an arbitrary element a_{md} from the equivalence class A_{md}. Let us apply the same technique as in the proof of Theorem 3.1, i.e., regroup the terms according to the number of errors s in the training sample:

$$Q_{\varepsilon,m} = \sum_{d=1}^{D_m}\sum_{s=s_0}^{\min\{l,m\}}\frac{1}{N}\sum_{n=1}^{N}[a_n \in A_{md}]$$

$$\times\left[\nu(a_{md}, X_n^l) = \frac{s}{l}\right]\left[\frac{m-s}{k} - \frac{s}{l} \geq \varepsilon\right]$$

$$= \sum_{d=1}^{D_m}\sum_{s=s_0}^{s_m^-(\varepsilon)}\underbrace{\frac{1}{N}\sum_{n=1}^{N}[a_n \in A_{md}]\left[\nu(a_{md}, X_n^l) = \frac{s}{l}\right]}_{\gamma(m,\,s)}.$$

Let us obtain an upper bound for the inner sum $\gamma(m, s)$, replacing $[a_n \in A_{md}]$ by unity. Reasoning along the same lines as in the proof of Theorem 3.1, we obtain $\gamma(m, s) \leq h\begin{pmatrix} l & s \\ L & m\end{pmatrix}$. This quantity does not depend on d; therefore, we can factor it out from the sum over d:

$$Q_{\varepsilon,m} \leq D_m \sum_{s=s_0}^{\varepsilon_m^-(\varepsilon)} h\begin{pmatrix} l & s \\ L & m\end{pmatrix} = D_m H_L^{l,m}(s_m^-(\varepsilon)).$$

The theorem is proved.

Theorem 4.3. The bound $Q_\varepsilon \leq \Delta_L^l \Gamma_L^l(\varepsilon)$ holds for any $\varepsilon \in [0, 1)$.

Proof. The proof follows immediately from the previous theorem:

$$\begin{aligned} Q_\varepsilon &= \sum_{m=0}^{L} Q_{\varepsilon,m} \leq \sum_{m=0}^{L} D_m H_L^{l,m}(s_m^-(\varepsilon)) \\ &\leq \left(\sum_{m=0}^{L} D_m\right)\max_m H_L^{l,m}(s_m^-(\varepsilon)) = \Delta_L^l \Gamma_L^l(\varepsilon). \end{aligned} \tag{4.10}$$

The theorem is proved.

Bound (4.10) differs from (4.7) by a factor of Δ_L^l; i.e., the confidence of prediction may become worse compared to the law of large numbers by a factor equal to the number of classes of distinguishable functions contained in the set A_L^l. As is shown in the proof, this bound may be significantly overestimated.

In a specific problem, the target function $y^*(x)$, the training sample X^l, and the learning algorithm μ are fixed. Therefore, as a result of learning, one can obtain only those functions from the set A that are considered appropriate for this problem by the learning algorithm μ. Other functions remain idle. This effect is called the *localization of a set of functions.* One should not necessarily restrict the complexity of the set to guarantee the learnability. It suffices to apply a learning algorithm capable of adapting to a problem by choosing an appropriate local subset $A_L^l(\mu, X^L)$ in the set A. This property of *localization ability* of an algorithm μ is an important ingredient of its generalization ability.

The classical VCT bound (4.4) is obtained by applying the operation of mathematical expectation E_{X^L} to (4.10), estimating from above the mathematical expectation of the shatter coefficient by the growth function, and estimating the hypergeometric tail by an exponential function:

$$\begin{aligned} \mathrm{E}_{X^L} Q_\varepsilon(\mu, X^L) &= \mathrm{E}_{X^L}\mathrm{P}_n[\delta_n \geq \varepsilon] \leq \mathrm{E}_{X^L}\Delta_L^l(\mu, X^L) \\ &\times \Gamma_L^l(\varepsilon) \leq \Delta^A(2l)\cdot\frac{3}{2}e^{-\varepsilon^2 l}, \end{aligned}$$

where the last inequality is valid under the assumption that $l = k$.

On the other hand, by analogy with Theorems 4.2 and 4.3, one can easily prove a full analog of the Vapnik–Chervonenkis bound (4.4) under the weak axiom:

$$\begin{aligned} P_\varepsilon(A) &= \mathrm{P}_n[\max_{a \in A}\delta(a, X_n^l, X_n^k) \geq \varepsilon] \\ &\leq \Delta^A(L)\Gamma_L^l(\varepsilon) \leq \Delta^A(2l)\cdot\frac{3}{2}e^{-\varepsilon^2 l}. \end{aligned} \tag{4.11}$$

Thus, the upper bounds of the functionals $P_\varepsilon(A)$ and $Q_\varepsilon(\mu, X^L)$ coincide. However, the functional Q_ε more accurately formalizes the concept of learnability.

Note that the idea of the proof of Theorems 4.2 and 4.3 is largely the same as that of Theorem P2 in [9, p. 221]; however, here the proofs are cleared of redundant probabilistic assumptions. It is these theorems that contain the core of the VCT. Many concepts and constructions of the VCT turn out to be redundant under the weak axiom. These are the set of functions, the uniform convergence of frequency to probability, the uniform convergence of frequencies in two subsamples, the main lemma (4.2), and the necessary conditions for uniform convergence. Bounds (4.10) and (4.9) are still overestimated; there are two reasons for this.

First, the shatter coefficient does not take into account the degree of similarity between the functions. Two indistinguishable functions make a total contribution of 1 to Δ_L^l. Two functions that are distinguishable on a half of objects of the sample X^L make a total contribution of 2. Two functions that are distinguishable only on one object also make a total contribution of 2, although this situation is much closer to the case of indistinguishable functions. As a rule, learning on similar subsamples X_n^l results in many similar functions. Each of these functions makes a contribution of 1 to Δ_L^l, which leads to the overestimated value of the shatter coefficient.

Second, the shatter coefficient does not take into account that the functions obtained as a result of learning are not equiprobable. Denote by N_{md} a subset of partitions for which one obtains functions from the equivalence class A_{md}:

$$N_{md} = \{n \in \{1, \dots, N\} | a_n \in A_{md}\}.$$

If $|N_{md}| \gg 1$, then functions from the class A_{md} are called *typical.* If the cardinality $|N_{md}|$ is close to unity, then the functions from the class A_{md} are called *atypical.* Most probably, there are functions of relatively law quality among them, which are obtained under nonrandom partitions of the sample, although the fraction of these functions is small and is bounded by the comfidence η. However, the number of these functions may turn out to be comparable with the local shatter coefficient. Each atypical function makes a contribution of 1 to Δ_L^l, although these functions might not need to be taken into account at all.

5. EMPIRICAL ANALYSIS OF BOUNDS OVERESTIMATION

The main factors responsible for VCT bounds overestimation can be seen from the proofs of Theorems 4.2 and 4.3. The weak axiom allows one to estimate the contribution of each of these factors empirically, by measuring functionals Q_ε and $Q_{\varepsilon, m}$ via cross-validation. Let $N' \subset \{1, \dots, N\}$ be a subset of partitions and $\hat{\mathrm{P}}_n \equiv \frac{1}{N'}\sum\limits_{n \in N'}$ be an empirical estimate of probability. Accordingly,

$$\hat{Q}_\varepsilon = \hat{\mathrm{P}}_n[\delta_n \geq \varepsilon],$$

$$\hat{Q}_{\varepsilon, m} = \hat{\mathrm{P}}_n[\delta_n \geq \varepsilon][\nu(a_n, X^L) = m/L],$$

where $a_n = \mu X_n^l$ is a function learned by an algorithm μ from the training subsample X_n^l and $\delta_n = \nu(a_n, X_n^k) - \nu(a_n, X_n^l)$ is its overfitting.

The question arises: what values should the local shatter profile take in order that bound (4.9) be not overestimated?

Definition 5.1. The sequence of values

$$\hat{D}_m(\varepsilon) = \frac{\hat{Q}_{\varepsilon, m}}{H_L^{l, m}(s_m^-(\varepsilon))}, \quad m = 0, \dots, L,$$

is called an *effective local shatter profile.*

Definition 5.2. The value $\hat{\Delta}_L^l(\varepsilon) = \hat{D}_0(\varepsilon) + \hat{D}_1(\varepsilon) + \dots + \hat{D}_L(\varepsilon)$ is called an *effective local shatter coefficient.*

This is an inverse problem: knowing empirical estimates of functionals $\hat{Q}_{\varepsilon, m}$ and $\hat{Q}_\varepsilon$, we should estimate the shatter profile D_m and the shatter coefficient Δ_L^l. Naturally, such estimates cannot be used for solving the main (direct) problem. The reason why they are introduced is to separate and compare various factors responsible for overestimation and to show what values of the shatter coefficients should be obtained theoretically.

Effective shatter coefficients may take noninteger values. Moreover, they depend on the accuracy ε. For a reasonable choice of ε, we first define the value of confidence η_0 or the range $[\eta_1, \eta_2]$ (in our experiments, these are 0.05 and [0.01, 0.1]). The accuracy and confidence are related by a nonincreasing function $\eta(\varepsilon) = Q_\varepsilon \approx \hat{Q}_\varepsilon$. This fact allows us to calculate an appropriate value of accuracy $\varepsilon = \eta^{-1}(\eta_0)$ or the range of values of accuracy $[\varepsilon_1, \varepsilon_2] = [\eta^{-1}(\eta_2), \eta^{-1}(\eta_1)]$, which, in turn, determine the range of the shatter coefficient:

$$\hat{\Delta}_L^l \in [\min_{\varepsilon \in [\varepsilon_1, \varepsilon_2]} \hat{\Delta}_L^l(\varepsilon), \max_{\varepsilon \in [\varepsilon_1, \varepsilon_2]} \hat{\Delta}_L^l(\varepsilon)].$$

Now, consider the main factors responsible for overestimation and the methods of their empirical measurement. The *degree of overestimation* is a ratio indicating how much the upper bound is overestimated.

1. Neglect of the localization effect. A local shatter coefficient Δ_L^l may turn out to be much less than the global shatter coefficient $\Delta(A, X^L)$ and, especially, the growth function $\Delta^A(L)$. The degree of overestimation can be calculated as the ratio

$$r_1 = \frac{\Delta^A(L)}{\Delta_L^l},$$

provided that both a theoretical bound of the growth function $\Delta^A(L)$ and the local coefficient Δ_L^l are known. The number of partitions $|N'|$ is a trivial and, as a rule, strongly underestimated bound of Δ_L^l.

2. Factorization of the shatter coefficient. In the proof of Theorem 4.2, the upper bound is calculated only once, to factor out the shatter coefficient D_m. The effective local profile $\hat{D}_m(\varepsilon)$ determines the values of the factors D_m necessary for the bound not to be overestimated. The degree of overestimation is determined by the ratio

$$r_2(\varepsilon) = \frac{\Delta_L^l}{\hat{\Delta}_L^l(\varepsilon)}.$$

3. Convolution of the shatter profile $\{D_m\}_{m=0}^{L}$ into a scalar complexity characteristic—the shatter coefficient $\Delta_L^l = \sum_{m=0}^{L} D_m$. This step was made when proving Theorem 4.3. The degree of overestimation is determined by the ratio

$$r_3(\varepsilon) = \frac{\sum_{m=0}^{L} \hat{D}_m(\varepsilon)\Gamma_L^l(\varepsilon)}{\sum_{m=0}^{L} \hat{D}_m(\varepsilon)H_L^{l,m}(s_m^-(\varepsilon))} = \frac{\hat{\Delta}_L^l(\varepsilon)\Gamma_L^l(\varepsilon)}{\hat{Q}_\varepsilon}.$$

4. Exponential approximation of a hypergeometric tail is motivated only by a desire to obtain a more elegant formula. When all the calculations are performed on a computer, the exponential approximation becomes inexpedient. The degree of overestimation is determined by the ratio

$$r_4(\varepsilon) = \frac{1.5e^{-\varepsilon^2 l}}{\Gamma_L^l(\varepsilon)}.$$

The product of the ratios obtained gives the degree of overestimation of the VCT bound:

$$r_1 \cdot r_2(\varepsilon) \cdot r_3(\varepsilon) \cdot r_4(\varepsilon) = \frac{\Delta^A(L) \cdot 1.5e^{-\varepsilon^2 l}}{\hat{Q}_\varepsilon}.$$

Effective VC dimension. The effect of localization is caused by fixing a target function y^*, a learning algorithm μ, and a sample X^L. In [11, 12], the concept of *effective VC dimension* is introduced, which takes into consideration μ and X^L but does not take into consideration y^*. Hence, the ratio of the effective growth function to the effective local shatter coefficient gives the degree of overestimation r_1 related only to the target function y^*.

Following [11], we restrict ourselves to a classification problem with two classes, $\mathbb{Y} = \{0, 1\}$, when the loss function is $\mathcal{L}(y, y') = [y \neq y']$ and $l = k$.

An *effective growth function* is defined as the value of $\Delta^A(L)$ for which bound (4.11) becomes exact (not overestimated):

$$\hat{\Delta}_{\text{eff}}^A(L, \varepsilon) = \frac{1}{\Gamma_L^l(\varepsilon)}\hat{\mathsf{P}}_n[\max_{a \in A}\delta(a, X_n^l, X_n^k) \geq \varepsilon].$$

An *effective VC dimension h* is defined as a parameter related to the effective growth function by formula (4.5): $\hat{\Delta}_{\text{eff}}^A(L) = 1.5L^h/h!$. To measure h, the authors of [11] suggest that one should estimate $\hat{\Delta}_{\text{eff}}^A$ for various L and then choose a value of h such that the function $1.5L^h/h!$ provides the most accurate approximation of the function $\hat{\Delta}_{\text{eff}}^A(L)$. Sufficiently high accuracy of approximation obtained in [11] indicates that the method works properly.

A search for a function $\tilde{a}_n \in A$ that maximizes $\delta(a, X_n^l, X_n^k)$ is equivalent to minimizing the empirical risk $\nu(a, \tilde{X}_n^L)$ using a modified full sample $\tilde{X}_n^L$: on all objects $x_i \in X_n^k$, a correct answer y_i is replaced by an erroneous answer $1 - y_i$:

$$\tilde{a}_n = \arg\max_{a \in A}\delta(a, X_n^l, X_n^k)$$

$$= \arg\min_{a \in A}\left(\frac{1}{l}\sum_{x_i \in X_n^l}[a(x_i) \neq y_i] + \frac{1}{k}\sum_{x_i \in X_n^k}[a(x_i) = y_i]\right).$$

To obtain a function $\tilde{a}_n$, one applies the same learning algorithm μ to the modified full sample $\tilde{X}_n^L$. The algorithm actually learns to make errors on a random half of objects. This removes the fixation of the target function y^* and the related part of the localization effect.

The degree of overestimation related to the neglect of the target function y^* is given by

$$r_1'(\varepsilon) = \frac{\hat{\mathsf{P}}_n[\delta(\tilde{a}_n, X_n^l, X_n^k) \geq \varepsilon]}{\hat{\mathsf{P}}_n[\delta(a_n, X_n^l, X_n^k) \geq \varepsilon]} = r_3(\varepsilon)\frac{\hat{\Delta}_{\text{eff}}^A(L, \varepsilon)}{\hat{\Delta}_L^l(\varepsilon)}.$$

Consider two interpretations of the coefficient $r_1'(\varepsilon)$.

1. Experiments with a linear threshold classifier described in [11] have given a quite expected result: the effective VC dimension is approximately equal to the dimension of the subspace in which the sample is concentrated. The coefficient $r_1'(\varepsilon)$ shows how much this bound is overestimated.

2. The *effective growth function* is determined in terms of the uniform convergence functional P_ε, which itself represents a certainly overestimated bound. The *effective local shatter coefficient* is determined in terms of the complete cross-validation functional Q_ε, which provides a more accurate formalization of the concept of learnability. This implies the second interpretation: $r_1'(\varepsilon)$ is the degree of overestimation caused by the application of the uniform convergence principle.

6. GENERALIZATION BOUNDS OF RULES

Rule induction classifiers are especially convenient for carrying out an empirical measurement of the degree of overestimation. First, for these classifiers, the growth function is well known. Second, they are based on an explicit search through a large number of elementary classifiers (rules), which allows one to efficiently estimate local shatter coefficients. Third, these classifiers are widely used in practice; therefore, the problem of overfitting of both the classifier itself and the rules constituting this classifier is of great practical interest.

Consider classification problems; let $\mathbb{Y}$ be a finite set of class labels.

A predicate $\varphi\colon \mathbb{X} \longrightarrow \{0, 1\}$ is said to *cover* an object x if $\varphi(x) = 1$. The predicate φ is characterized by two values with respect to a class $y \in \mathbb{Y}$ and a sample X^l: the number of positive examples p_y (covered objects of class y) and the number of negative examples b_y (covered objects of other classes y):

$$p_y(\varphi_y, X^l) = \#\{x_i \in X^l \,|\, \varphi_y(x_i) = 1, y_i = y\},$$

$$b_y(\varphi_y, X^l) = \#\{x_i \in X^l \,|\, \varphi_y(x_i) = 1, y_i \neq y\}.$$

A *rule* of a class $y \in \mathbb{Y}$ is a predicate $\varphi_y\colon X \longrightarrow \{0, 1\}$ that covers a sufficiently large number of objects of class y and a sufficiently small number of objects of all the other classes: $p_y(\varphi_y, X^l) \geq p_{y0}$ and $b_y(\varphi_y, X^l) \leq b_{y0}$, where p_{y0} and b_{y0} are prescribed threshold constants.

The quality of a rule is characterized by a rule evaluation heuristic $I(p_y, b_y)$. In practice, the heuristic can be introduced in different ways; in particular, an entropy criterion of *information gain*, statistical criteria ξ^2 and ω^2, Fisher's exact test [13], boosting criterion $I(p_y, b_y) = \sqrt{p_y} - \sqrt{b_y}$ [14], and other criteria are applied.

A *rule-based classifier* is a linear combination of rules:

$$a(x) = \underset{y \in Y}{\operatorname{argmax}} \sum_{t=1}^{T_y} w_y^t \varphi_y^t(x),$$

where $\varphi_y^t(x)$ are rules from the class y, w_y^t are the weights of the rules, and T_y is the number of rules of class y. Many rule-based classifiers can be represented in this form: weighted voting of rules [14], decision lists [15], decision trees [16], set covering machines [17], etc.

It is convenient to measure empirically the degree of overestimation for rules rather than for classifiers. To this end, one should slightly change the basic definitions. The modifications are quite technical; the formulations of the main theorems remain virtually the same.

A *rule learning algorithm* of class y is a map μ_y that generates a set of rules from the training sample X^l:

$$\mu_y X^l = \{\varphi_y^t(x) \,|\, t = 1, \ldots, T_y\}.$$

The *error rate* of a rule φ_y on a sample X^l is given by

$$\begin{aligned} \nu_y(\varphi_y, X^l) &= \frac{1}{l}\sum_{i=1}^{l} [\varphi_y(x_i) \neq [y_i = y]] \\ &= \frac{1}{l}(b_y(\varphi_y, X^l) + P_y - p_y(\varphi_y, X^l)), \end{aligned} \tag{6.1}$$

where P_y is the number of objects of class y on the sample X^l. When $\varphi_y(x_i) = 0$ and $y_i = y$, the rule makes an error of kind I: it does not cover an object of a positive class. When $\varphi_y(x_i) = 1$ and $y_i \neq y$, the rule makes an error of kind II: it covers an object of a negative class. Usually, errors of kind I are less dangerous, because a missing object can be covered by other rules.

The *overfitting* of a rule $\varphi \in \mu_y X^l$ for a given test sample X^k is the difference of its error rates on the test and training samples:

$$\delta_y(\varphi, X^l, X^k) = \nu_y(\varphi, X^k) - \nu_y(\varphi, X^l).$$

The functional of complete cross-validation $Q_\varepsilon(\mu_y, X^L)$ is defined as a part of overfitted rules, among all rules of class y, generated by the rule learning algorithm μ_y on all possible subsamples $X_n^l \subset X^L$:

$$Q_\varepsilon(\mu_y, X^L) = \mathsf{P}_n \frac{1}{|\mu_y X_n^l|} \sum_{\varphi \in \mu_y X_n^l} [\delta_y(\varphi, X_n^l, X_n^k) \geq \varepsilon].$$

Predicates $\varphi, \varphi'\colon X \longrightarrow \{0, 1\}$ are said to be *indistinguishable*, or equivalent, on a sample X^L if $\varphi(x) = \varphi'(x)$ for any $x \in X^L$. The *shatter coefficient* $\Delta(\Phi, X^L)$ of a set of predicates Φ on a sample X^L is the number of equivalence classes induced on Φ by the indistinguishability relation. Consider the set of rules obtained by the algorithm μ_y on all possible training subsamples: $\Phi_L^l = \bigcup_{n=1}^{N} \mu_y X_n^l$. The shatter coefficient $\Delta_L^l = \Delta(\Phi_L^l, X^L)$ of this set is called a *local shatter coefficient* of the algorithm μ_y on the sample X^L. The set of rules Φ_L^l is partitioned into $L + 1$ subsets Φ_m that consist of rules with a fixed number m of errors on the full sample X^L:

$$\Phi_m = \left\{\varphi \in \Phi_L^l \,\middle|\, \nu_y(\varphi, X^L) = \frac{m}{L}\right\}, \quad m = 0, \ldots, L.$$

A *local shatter profile* of the algorithm μ_y on the sample X^L is a sequence of shatter coefficients $D_m = \Delta(\Phi_m, X^L)$, $m = 0, \ldots, L$.

It is obvious that $\Delta_L^l = D_0 + \ldots + D_L$.

Along with the functional Q_ε, we define a functional $Q_{\varepsilon, m}$ as a part of overfitted rules that make m errors on X^L:

$$\begin{aligned} &Q_{\varepsilon, m}(\mu_y, X^L) \\ &= \mathsf{P}_n \frac{1}{|\mu_y X_n^l|} \sum_{\varphi \in \mu_y X_n^l} [\delta_y(\varphi, X_n^l, X_n^k) \geq \varepsilon]\left[\nu_y(\varphi, X^L) = \frac{m}{L}\right]. \end{aligned}$$

In this notation, Theorems 4.2 and 4.3 remain valid for the case of rules. The modification has mainly concerned the definition of a local set of functions A_L^l; now the role of this set is played by the local set of rules Φ_L^l. The meaning of the modification is quite simple: one should take into consideration all the rules $\varphi_y^t(x)$ learned from all partitions n. The method of empirical measurement of Q_ε, $Q_{\varepsilon,m}$, $\hat{\Delta}_L^l$, and $\hat{D}_m$ remains the same.

Note that all these quantities are determined for each class $y \in \mathbb{Y}$ separately and may be different for different classes.

The method proposed essentially refines the earlier variants [18, 19].

The **rule learning algorithm** applied in our experiments is based on three essential heuristics: breadth-first search [20], boosting the rules [14], and Fisher's exact test as a rule evaluation criterion [13]. The algorithm was realized by D. Kochedykov and A. Ivakhnenko and is applied in the Forecsys ScoringAce® system [21, 18, 19]. Here we present a simplified description of this algorithm.

Suppose that objects $x \in \mathbb{X}$ are described by n discrete features f_j: $\mathbb{X} \longrightarrow D_j$, $j = 1, \dots, n$. Nominal features give rise to *elementary predicates (terms)* of two types: $\beta_j(x) = [f_j(x) = c]$ and $\beta_j(x) = [f_j(x) \neq c]$ for all possible $c \in D_j$. In addition, order features generate two more types of terms: $\beta_j(x) = [f_j(x) \le c]$ and $\beta_j(x) = [f_j(x) \ge c]$, $c \in D_j$. Denote by $\mathcal{B}_j$ the set of all terms generated by a feature f_j. The search of rules is performed among conjunctions of rank at most K that are composed of terms:

$$\Phi[K] = \left\{ \varphi(x) = \bigwedge_{j \in J} \beta_j(x) \middle| \beta_j \in \mathcal{B}_j, J \subseteq \{1, \dots, n\}, |J| \le K \right\}.$$

Algorithm 6.1 starts the search from a set of conjunctions of rank 1. To this end, at most T_1 terms are chosen that have the best values of evaluation criterion. At all subsequent steps, one term is added to each conjunction in all possible ways. Again, at most T_1 conjunctions from this extended set are chosen that have the best values of evaluation criterion. The extension of conjunctions stops either on reaching the maximal rank K or when none of the conjunctions can be improved by adding a term. The best conjunctions collected from all steps are included in the lists R_y. The parameter T_1 controls the *breadth of the search* and allows one to trade off between the quality of rules and the time efficiency of the algorithm.

The quality of a predicate $\varphi(x)$ with respect to the training sample X^l and the class y is evaluated by two criteria: a part of erroneously covered objects $E_y(\varphi) = \dfrac{b_y}{p_y + b_y}$ and the informativity $I_y(\varphi) = \ln C_{P_y+B_y}^{p_y+b_y} - \ln C_{P_y}^{p_y} C_{B_y}^{b_y}$, where P_y is the number of positive objects and B_y is the number of negative objects in X^l.

After a run of Algorithm 6.1, there may remain objects in the sample that either have not been covered by any rule from the lists R_y or have been erroneously covered by the rules of wrong classes. These objects receive larger weights according to the boosting formula [14], and Algorithm 6.1 is restarted. The weights of objects are taken into consideration when calculating the rule evaluation criterion $I_y(\varphi)$, which allows one to find new rules that essentially differ from those found at previous iterations.

Algorithm 6.1. Learning conjunctions by a breadth-first search algorithm

Input:

X^l is a training sample, $y \in \mathbb{Y}$ is a class for which conjunctions are constructed, K is the maximal rank of conjunctions, T_1 is the number of best conjunctions chosen at each step, T_0 is the number of best conjunctions chosen at the last step, $I_{\min}$ is the informativity threshold, and $E_{\max}$ is the admissible number of errors;

Output:

the list of conjunctions $R_y = \{\varphi_y^t(x) | t = 1, \dots, T_y\}$;

```
$R_y := \varnothing$;
for any $\beta \in \mathcal{B}_j, j = 1, \dots, n$
    addtothelist ($R_y$, $\beta$);
for any $k = 2, \dots, K$
    for any conjunctions $\varphi \in R_y$ of rank $(k - 1)$
        for any $\beta \in \mathcal{B}_j, j = 1, \dots, n$
            if a feature $f_j$ is not used in conjunction $\varphi$ and
            $I_y(\varphi \wedge \beta) \ge I_{\min}$, then
                addtothelist ($R_y$, $\varphi \wedge \beta$);
leave at most $T_0$ conjunctions with the maximal $I_y(\varphi)$
and $E_y(\varphi) \le E_{\max}$ in $R_y$;

PROCEDURE addtothelist ($R_y$, $\varphi$);
if $|R_y| < T_1$, then
    $R_y := R_y \cup \{\varphi\}$
else
    find the worst conjunction in the list: $\psi :=$
    $\arg\min_{\psi \in R_y} I_y(\psi)$;
    if $I_y(\varphi) > I_y(\psi)$, then
        replace the worst conjunction $\psi$ by $\varphi$ in the list $R_y$.
```

Table 1. Characteristics of problems: sample length L; number of features n; the number of generated terms d_j, where the expression 20^5 indicates that there are five features each generating 20 terms; test error (in percent) for four standard algorithms according to [22, 14]; and test error rate for Algorithm 6.1

Problem	L	n	$d_1 \ldots d_n$	C4.5	C5.0	RIPPER	SLIPPER	Forecsys
crx	690	15	$2^4 3^2 4^1 9^1 14^1 20^6$	15.5	14.0	15.2	15.7	14.3 ± 0.2
german	1000	20	$2^2 3^3 4^3 5^5 10^1 11^1 20^5$	27.0	28.3	28.7	27.2	28.5 ± 1.0
hepatitis	155	19	$2^{13} 6^4 8^1 9^1$	18.8	20.1	23.2	17.4	16.7 ± 1.7
horse-colic	300	25	$2^3 3^2 4^6 5^5 6^2 20^7$	16.0	15.3	16.3	15.0	16.4 ± 0.5
hypothyroid	3163	25	$2^{18} 20^7$	0.4	0.4	0.9	0.7	0.8 ± 0.04
liver	345	6	$12^1 20^5$	37.5	31.9	31.3	32.2	29.2 ± 1.6
promoters	106	57	57^4	18.1	22.7	19.0	18.9	12.0 ± 2.0

The growth function $\Delta^{\Phi[K]}(L)$ of the set $\Phi[K]$ does not exceed its cardinality. Suppose that the jth feature generates $d_j = |\mathcal{B}_j|$ terms, $j = 1, \ldots, n$. Then the number of conjunctions of rank r constructed from the features of the subset $J = \{1, \ldots, j\}$ does not exceed

$$H_{r,j} = \sum_{\substack{J' \subseteq J \\ |J'| = r}} \prod_{j \in J'} d_j.$$

The numbers $H_{r,j}$ can be calculated effectively, in $O(Kn)$ operations, if one applies the following recurrence formulas: $H_{0,j} = 1$, $H_{r,j} = 0$ for $r > j$, and

$$H_{r,j+1} = H_{r,j} + d_j H_{r-1,j},$$

$$j = 1, \ldots, n, \quad r = 1, \ldots, K.$$

The growth function does not exceed the total number of conjunctions with ranks from 1 to K:

$$\Delta^{\Phi[K]}(L) \le H_{1,n} + \ldots + H_{K,n}.$$

Local shatter coefficient Δ_L^l is estimated by the total number of conjunctions that fall into the lists R_y over all training samples X_n^l, $n \in N'$:

$$\underline{\underline{\Delta}}_L^l = \sum_{n \in N'} \left|\mu_y X_n^l\right| \le |N'| T_0.$$

This bound may be underestimated since $|N'| \ll N$. A more adequate bound is given by the number $\underline{\Delta}_L^l$ of analyzed conjunctions that satisfy the criteria of high informativity $I_y(\varphi) \ge I_{\min}$ and low errors $E_y(\varphi) \le E_{\max}$. This number can easily be calculated during the search process.

Denote by $\overline{\Delta}_L^l$ the number of all conjunctions φ for which one calculates the characteristics $p_y(\varphi, X^l)$ and $b_y(\varphi, X^l)$ during the search. A trivial and slightly overestimated bound is given by $\overline{\Delta}_L^l \le |N'|(T_1 K - T_1 + 1)(d_1 + \ldots + d_n)$. The exact number of all analyzed conjunctions can also be calculated easily during the search.

For Algorithm 6.1, there is another way to estimate the degree of overestimation related to the localization of the target function y^*. This is the ratio of the number of all analyzed conjunctions to the number of conjunctions that turned out to be rules:

$$r_1''(\varepsilon) = \frac{\overline{\Delta}_L^l}{\underline{\Delta}_L^l}.$$

Since Algorithm 6.1 performs a directed search of the best conjunctions, this ratio may be slightly underestimated.

7. EXPERIMENTS, RESULTS, AND CONCLUSIONS

The rule learning algorithm was tested on seven two-class classification problems from the UCI repository [23]. The sample was partitioned randomly 20 times into two equal parts, $l = k$, with stratification of classes. In each partitioning, the first half of the sample was used as a training sample and the second half as a test sample; then, these halves changed places. Thus, $|N'| = 40$. Table 1 shows the characteristics of the problems and the mean error on test data. The data on algorithms C4.5, C5.0, RIPPER, and SLIPPER are borrowed from [22, 14] and show that the quality of the algorithm implemented is comparable with that of its analogs (we do not aim to prove the advantages of our algorithm in this paper).

Table 2 shows the bounds for the shatter coefficients calculated during running Algorithm 6.1. The two right columns present the bounds for the effective local shatter coefficient calculated according to Definition 5.2.

Figure 1 represents the graphs of the coefficient $\hat{\Delta}_L^l$ as a function of accuracy ε. The decaying curve shows the confidence $\hat{Q}_\varepsilon$ as a function of ε. To determine the

Table 2. Parameters of the algorithm: search breadth T_1, maximal rank of conjunctions K, and the class label in UCI encoding. Bounds for shatter coefficients: growth function $|\Phi[K]|$, average number of analyzed conjunctions $\frac{1}{|N'|}\bar{\Delta}_L^l$, average number of informative conjunctions $\frac{1}{|N'|}\underline{\Delta}_L^l$, average number of conjunctions chosen by the rule learning algorithm $\frac{1}{|N'|}\underline{\underline{\Delta}}_L^l$, and effective local shatter coefficient $\hat{\Delta}_L^l(\varepsilon)$ corresponding to the range of $\hat{Q}_\varepsilon \in [0.01, 0.1]$ and a value of $\hat{Q}_\varepsilon = 0.05$

Problem	T_1	K	y	$\lvert\Phi[K]\rvert$	$\frac{1}{\lvert N'\rvert}\bar{\Delta}_L^l$	$\frac{1}{\lvert N'\rvert}\underline{\Delta}_L^l$	$\frac{1}{\lvert N'\rvert}\underline{\underline{\Delta}}_L^l$	$\hat{\Delta}_L^l[\varepsilon_1, \varepsilon_2]$	$\hat{\Delta}_L^l(\varepsilon_0)$
crx	50	4	0	1.4×10^7	2.1×10^4	380	5	[10; 41]	24
			1			490	6	[11; 180]	12
german	50	5	1	5.2×10^8	3.0×10^4	1370	14	[38; 530]	54
			2			330	3	[1.0; 2.2]	1.9
hepatitis	50	4	0	5.6×10^5	0.9×10^4	570	7	[11; 148]	83
			1			240	3	[12; 27]	15
horse-colic	50	5	1	1.9×10^6	3.8×10^4	630	7	[2; 9]	7
			2			330	3	[3; 6]	6
hypothyroid	100	5	0	5.3×10^8	6.3×10^4	210	7	[3; 220]	21
			1			80	3	[2; 44]	30
liver	50	4	0	1.9×10^6	1.1×10^4	700	7	[4; 21]	12
			1			650	7	[3; 12]	5
promoters	50	3	0	1.0×10^8	2.2×10^4	480	5	[36; 230]	72
			1			300	3	[9; 22]	18

range of possible values of $\hat{\Delta}_L^l(\varepsilon)$, we first fix the range of reasonable values of confidence $\hat{Q}_\varepsilon \in [0.01, 0.1]$ (on the right vertical axis); for this range, we determine the range of accuracy (on the horizontal axis), and, on this range, we determine the minimal and maximal values of $\hat{\Delta}_L^l(\varepsilon)$.

Table 3 presents the bounds for the degrees of overestimation calculated for a fixed value of confidence of $\hat{Q}_\varepsilon = 0.05$.

Interpretation and conclusions. Among four factors responsible for overestimation, the first two prove to be most significant: r_1, neglect of the localization effect, and r_2, factorization of a shatter coefficient. A large amount of work on *data-dependent bounds* have been devoted to the elimination of the first factor [24, 25, 6]. However, all these bounds contain a multiplier that describes the complexity of a certain set of functions, even though a local set. The large values of r_2 indicate that the "curse of overestimation" is inherent in all complexity bounds.

The coefficients r_1' and r_1'' estimate the contribution of the target function y^* localization to the degree of overestimation of r_1. Both these coefficients are underestimated; therefore, we can argue that the corresponding loss of accuracy amounts to two orders of magnitude or greater. The concept of *effective dimension* introduced by Vapnik does not take into account this factor because it is based on the uniform convergence principle.

In all problems, the effective local shatter coefficient does not exceed the sample length L. Attempts to use this coefficient for determining the *effective local dimension* by formula (4.5) lead to a degenerate result: in practice, such a dimension does not exceed one. This again means that complexity bounds (even local ones) are intrinsically extremely overestimated. To substantiate the learnability, one should introduce some other, much finer, characteristics of a learning algorithm.

The factor r_3 is relatively small in most cases. For the error numbers $m = L\nu_y(\varphi, X^L)$, typical for rules, the values of $H(m) = H_L^{l,m}(s_m^-(\varepsilon))$ are close to the maximum (see Fig. 2). However, as $m \longrightarrow 0$, the function $H(m)$ tends to zero faster than a geometric progression. Therefore, for conventional classifiers and "good" problems with error rate (approximately) less than 10%, the factor r_3 may reach considerable values.

The factor r_4 shows that the exponential approximation of the hypergeometric tail is loose and should not be used in practice.

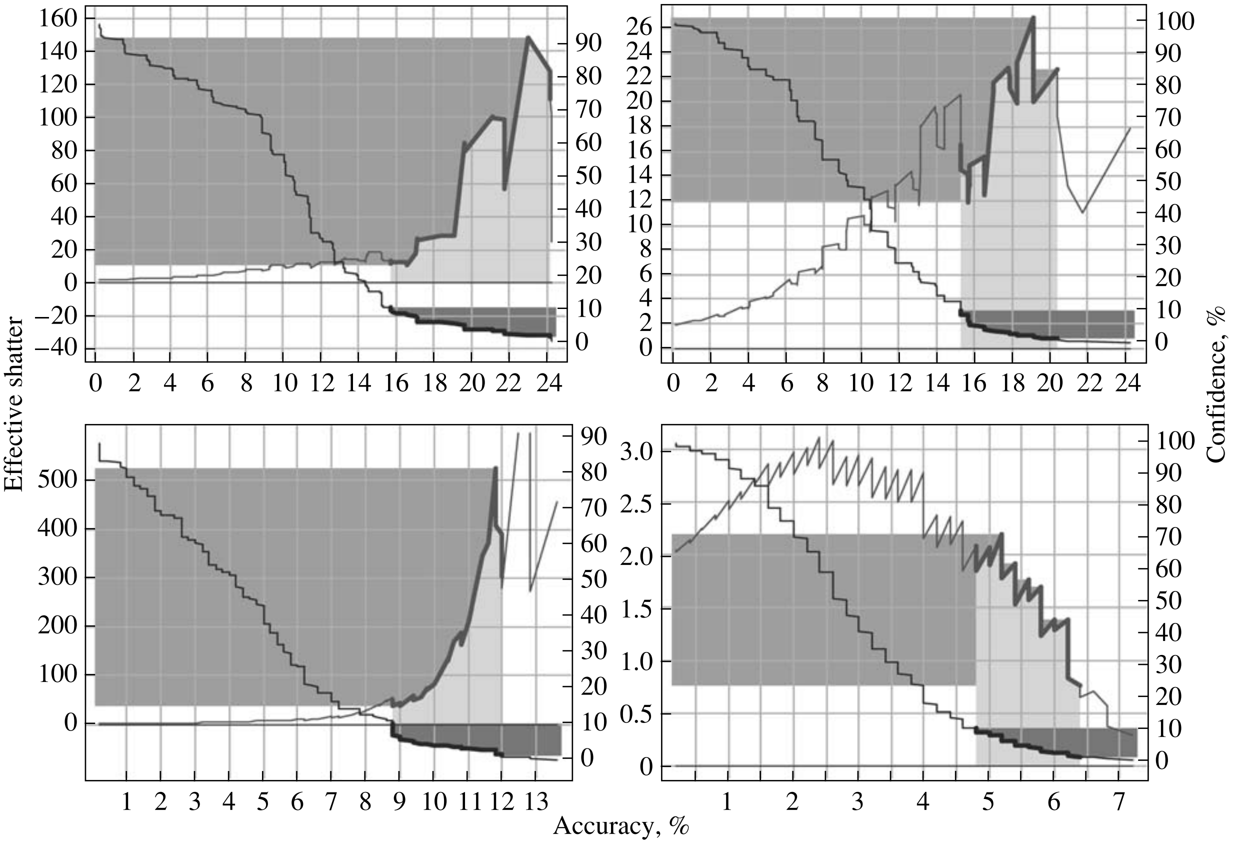

Fig. 1. Effective local shatter coefficient $\hat{\Delta}_L^l$ and confidence $\hat{Q}_\varepsilon$ as functions of accuracy ε for problems hepatitis (top figures, $y = 0.\ 1$) and german (bottom figures, $y = 1, 2$). The strips indicate the determination of the range of possible values of $\hat{\Delta}_L^l$ by a given range of confidence $\hat{Q}_\varepsilon \in [0.01, 0.1]$.

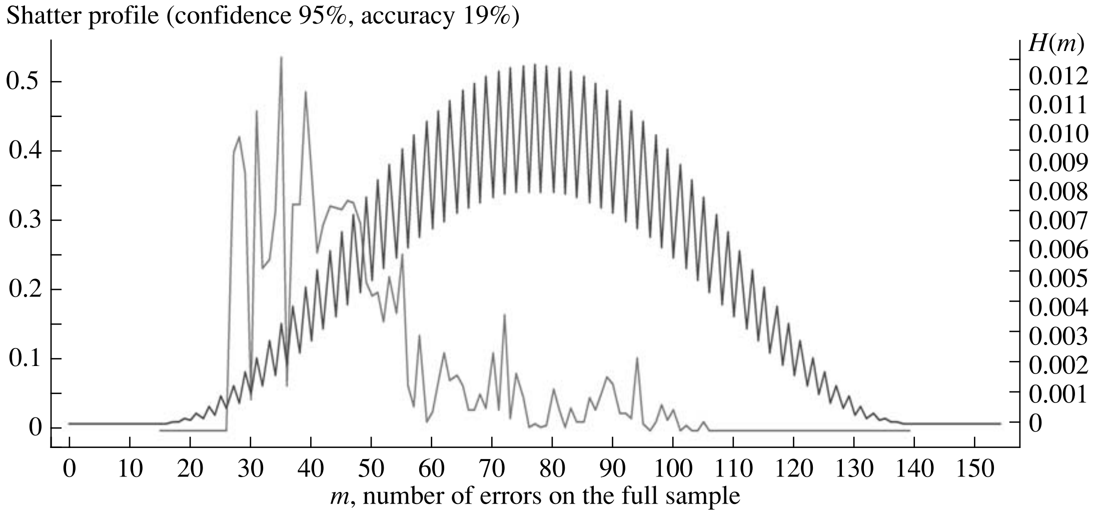

Fig. 2. Effective local profile $\hat{D}_m(\varepsilon)$ and function $H(m)$ versus the number of errors $m = L\nu_y(\varphi, X^L)$ for problem hepatitis, $y = 0$.

Table 3. Degrees of overestimation for the values of accuracy ε corresponding to a value of confidence of $\hat{Q}_\varepsilon = 0.05$

Problem	y	r_1	$r_1'(\varepsilon)$	$r_1''(\varepsilon)$	$r_2(\varepsilon)$	$r_3(\varepsilon)$	$r_4(\varepsilon)$
crx	0	890	20	55	680	3.1	32.6
	1	690	21	43	1700	1.6	11.6
german	1	8950	18	22	1500	1.7	10.9
	2	37000	22	92	9000	1.2	9.9
hepatitis	0	23	20	16	280	13.4	9.5
	1	55	20	37	680	2.4	22.5
horse-colic	1	72	19	60	4500	2.1	7.2
	2	140	20	115	3400	3.6	7.3
hypothyroid	0	61000	21	310	400	32.2	16.5
	1	153000	15	770	460	3.8	28.7
promoters	0	94	16	46	340	5.9	9.8
	1	150	23	73	790	3.4	6.9

Conclusions and further work:

–Under the weak probabilistic axiom, the generalization bounds are obtained for the functionals based on complete cross-validation. This facilitates the empirical analysis of theoretical bounds and allows one to estimate empirically the factors responsible for the overestimation of bounds.

–It is interesting to apply the proposed empirical method to other learning algorithms to investigate their *localization ability.*

–When obtaining numerically tight generalization bounds, one should take into account not only the localization but also the nonuniformity of distribution and the degree of difference of algorithms. In tightest complexity bounds, the shatter coefficients would be on the order of 10^1–10^2.

ACKNOWLEDGMENTS

This work was supported by the Russian Foundation for Basic Research (project nos. 05-01-00877 and 08-07-00422), by the Program "Algebraic and Combinatorial Methods in Mathematical Cybernetics" of the Department of Mathematics, Russian Academy of Sciences, and by the Russian Science Support Foundation.

REFERENCES

1. A. N. Kolmogorov, *Information Theory and the Theory of Algorithms. Selected Works, Volume III,* Ed. by A.N. Shiryaev (Springer, 1993).
2. Yu. K. Belyaev, *Probabilistic Methods for Sampling Inspection* (Nauka, Moscow, 1975) [in Russian].
3. N. V. Smirnov, "A Bound for Discrepancy between Empirical Distribution Curves in Two Independent Samples," Byull. Mosk. Univ., Ser. A, No. 2, 3–14 (1939).
4. L. N. Bol'shev and N. V. Smirnov, *Tables of Mathematical Statistics* (Nauka, Moscow, 1983) [in Russian].
5. V. Vapnik, *Statistical Learning Theory* (Wiley, New York, 1998).
6. S. Boucheron, O. Bousquet, and G. Lugosi, "Theory of Classification: A Survey of Some Recent Advances," ESIAM: Probab. Stat., No. 9, 323–375 (2005).
7. K. V. Vorontsov, "Combinatorial Approach to Estimating the Quality of Learning Algorithms," in *Mathematical Problems of Cybernetics*, Ed. by O. B. Lupanov (Fizmatlit, Moscow, 2004), Vol. 13, pp. 5–36.
8. V. N. Vapnik and A. Ya. Chervonenkis, *Theory of Pattern Recognition* (Nauka, Moscow, 1974).
9. V. N. Vapnik, *Reconstruction of Functions from Empirical Data* (Nauka, Moscow, 1979) [in Russian].
10. V. N. Vapnik and A. Ya. Chervonenkis, "On the Uniform Convergence of the Frequencies of Events and Their Probabilities," Teor. Veroyatn. Ee Primen. **16** (2) (1971).
11. V. N. Vapnik, E. Levin, and Y. L. Cun, "Measuring the VC-Dimension of a Learning Machine," Neural Comput. **6** (5), 851–876 (1994).
12. L. Bottou, C. Cortes, and V. Vapnik, *On the Effective VC Dimension* (1994).
13. J. K. Martin, "An Exact Probability Metric for Decision Tree Splitting and Stopping," Machine Learning **28** (2–3) 257–291 (1997).
14. W. W. Cohen and Y. A. Singer, "A Simple, Fast and Effective Rule Learner," in *Proceedings of the 16th National Conference on Artificial Intelligence* (1999), pp. 335–342.
15. R. L. Rivest, "Learning Decision Trees," Machine Learning **2** (3), 229–246 (1987).
16. J. Quinlan, "Induction of Decision Trees," Machine Learning **1** (1), 81–106 (1986).

17. M. Marchand and J. Shawe-Taylor, "Learning with the Set Covering Machine," in *Proceedings of the 18th International Conference on Machine Learning* (Morgan Kaufmann, San Francisco, CA, 2001), pp. 345–352.
18. K. V. Vorontsov and A. A. Ivakhnenko, "Empirical Bounds of a Local Growth Function for Rule Induction Classifiers," in *Artificial Intellect* (2006), pp. 281–284 [in Russian].
19. A. A. Ivakhnenko and K. V. Vorontsov, "Upper Bounds for Overfitting and a Shatter Profile of Logical Rules," in *Mathematical Methods of Pattern Recognition* (MAKS, Moscow, 2007), pp. 33–37 [in Russian].
20. G. S. Lbov, *Methods for Processing Experimental Data of Different Types* (Nauka, Novosibirsk, 1981) [in Russian].
21. D. A. Kochedykov, A. A. Ivakhnenko and K. V. Vorontsov, "A Credit Scoring System Based on Logical Classification Algorithms," in *Mathematical Methods of Pattern Recognition-12* (MAKS, Moscow, 2005), pp. 349–353 [in Russian].
22. W. W. Cohen, "Fast Effective Rule Induction," in *Proceedings of the 12th International Conference on Machine Learning* (Morgan Kaufmann, Tahoe City, CA, 1995), pp. 115–123.
23. A. Asuncion and D. Newman, "UCI Machine Learning Repository," Tech. Rep.: University of California, Irvine, School of Information and Computer Sciences, 2007.
24. V. Koltchinskii and D. Panchenko, "Rademacher Processes and Bounding the Risk of Function Learning," in *High Dimensional Probability*, II, Ed. by D. E. Gine and J. Wellner (Birkhauser, 1999), pp. 443–457.
25. P. L. Bartlett, S. Mendelson, and P. Philips, "Local Complexities for Empirical Risk Minimization," in *COLT: 17th Annual Conference on Learning Theory*, Ed. by J. Shawe-Taylor and Y. Singer (Springer, 2004), pp. 270–284.

Konstantin Vorontsov. Born 1971. Graduated from the Faculty of Control and Applied Mathematics, Moscow Institute of Physics and Technology, in 1994. Received candidate's degree in 1999. Currently is with the Dorodnicyn Computing Centre, Russian Academy of Sciences. Deputy director for research of Forecsys company (www.forecsys.com). Scientific interests: computational learning theory, machine learning, data mining, probability theory, and combinatorics. Author of 40 papers. Homepage: www.ccas.ru/voron